SEFEL: A Simple Yet Effective Framework for Fast Event Linking

Yinan Liu[1], Ziyang Zhang[1], Bin Wang[1,2], Xiaochun Yang[1,2*]

[1]School of Computer Science and Engineering, Northeastern University, Shenyang, China
[2]National Frontiers Science Center for Industrial Intelligence and Systems optimization,
Northeastern University, Shenyang, China
liuyinan@cse.neu.edu.cn, 2472096@stu.neu.edu.cn, {binwang, yangxc}@mail.neu.edu

Abstract

Event linking aims to associate event mentions in text with their corresponding entries in a knowledge base (KB). This task can help text understanding to benefit downstream tasks (e.g., question answering) and expand the KB through new event knowledge mentioned in the text. Existing event linking approaches usually adopt a retrieve-and-rank framework, which suffers from high computational costs and relies on hand-crafted rules, thereby limiting generalization. Additionally, it is found that some entity linking methods can be used to solve this task directly. However, they also perform not well. In this paper, we propose SEFEL, an end-to-end, argument-aware event representation-based event linking framework to unify the modeling of both in-KB and out-of-KB scenarios. To further enhance the linking performance, we propose a contrastive learning module to refine the learned embeddings of events and event mentions. Experimental results demonstrate that SEFEL improves accuracy by at least 3.59 (in-KB) and 21.5 (out-of-KB) compared with baselines, while its inference speed is more than 38 times faster than baselines, showcasing its accuracy and efficiency.

Code — https://github.com/zyaa2266/SEFEL

Introduction

On the one hand, linking ambiguous event mentions in text to their corresponding events in an external KB can enhance the understanding of the text by utilizing the mapping events' background knowledge provided by the KB. Furthermore, incorporating such event-centric knowledge can enhance the performance of many downstream applications such as question answering (Li et al. 2024) and recommendation systems (Wang et al. 2022). On the other hand, texts mentioning new events embody key event knowledge, which is useful for enriching existing KBs (Hoffart, Altun, and Weikum 2014; Hoffart et al. 2016; Liu et al. 2020, 2023; Shen, Liu, and Wang 2018).

Despite its importance, the task of event linking faces several key challenges. (1) Diverse event expressions. The same event may be described in various ways in different texts, resulting in low lexical overlap and making the matching process more difficult. As the example shown in Figure 1, we can see that the event "Evacuation of East East Prussia" is described in two ways, with "evacuated" and "fled" as event mentions, respectively. (2) Cross-sentence event arguments. The semantics of an event rely heavily on contextual and its argument information (e.g., patients, time, and location). However, such information is often implicitly distributed across sentences. (3) Out-of-KB event identification. For the example in Figure 1, the event mention "evacuated" may refer to events "Evacuation of East Prussia" or "World War II" or an out-of-KB event (i.e., the event not in the KB). Due to the limited coverage of KBs, an effective event linking model must not only accurately link *in-KB* events but also robustly represent and identify *out-of-KB* events.

Recently, some studies have explored for the event linking task (Pratapa, Gupta, and Mitamura 2022; Yu et al. 2023; Hsu et al. 2024). They typically follow a two-stage retrieve-and-rank framework BLINK (Wu et al. 2020): a bi-encoder is used for candidate retrieval and a cross-encoder is used for fine-grained ranking over the retrieved candidate events with respect to the event mention. Yet, this architecture incurs a high computational cost during inference, since each event mention must be independently cross-encoded with multiple candidate events, making it difficult to meet the efficiency requirements of large-scale applications. Furthermore, among these methods, Hsu et al. (2024) achieves the best performance in terms of accuracy by incorporating event argument information into training. However, their event argument extraction method relies on hand-crafted rules tailored to specific data sets, which not only limit cross-domain generalization but also yield noisy or incomplete argument supervision, consequently impairing models' ability to identify *out-of-KB* events. In addition, once we neglect the event arguments, which is an intrinsic difference between entities and events, event linking task could be considered similar to the well-studied entity linking problem (Shen, Wang, and Han 2014). Unfortunately, existing entity linking methods fail to obtain good performance, which has been verified by our experiments.

To deal with the above issues, we propose a **S**imple yet **E**ffective **F**ast **E**vent **L**inking framework SEFEL. First, SEFEL leverages an LLM to extract event arguments from raw texts and generates samples for out-of-KB events through LLM-guided event argument substitution and contextual rewriting, effectively enriching the training data. Second,

*Xiaochun Yang is the corresponding author.

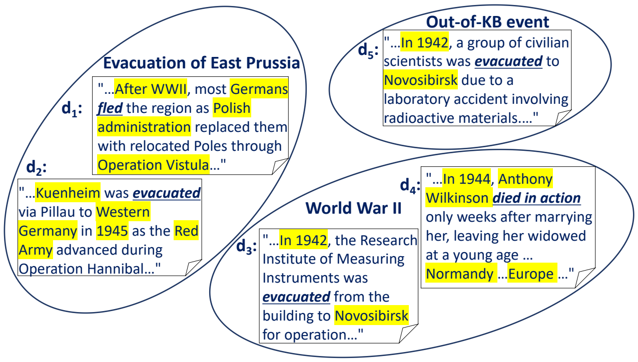

Figure 1: Some texts of different events. The event mentions are underlined and event arguments are highlighted.

SEFEL adopts a dual-tower transformer architecture to construct argument-aware event mention representations based on the LLM-extracted event arguments while encoding candidate events separately, projecting both into a shared embedding space. Specially, it introduces a learnable NIL embedding as a special candidate for each event mention, enabling explicit identification of out-of-KB events.

Better embeddings of event mentions and events contribute to better linking results. Intuitively, as shown in Figure 1, documents d_1 and d_2 referring to same event "Evacuation of East Prussia" should be clustered together in the embedding space, while d_1/d_2 and d_3/d_4 (describing different events "Evacuation of East Prussia" and "World War II") should be pushed apart. Additionally, for each event mention, the distance to its corresponding event should be minimized and the distance to other candidate events should be maximized. To achieve above, we propose a novel contrastive learning module from two views (i.e., mention-mention view and mention-candidate view) to boost the quality of the learned embeddings of event mentions and events and further enhance the linking performance. To our best knowledge, we are the first to employ contrastive learning for the event linking task.

The main contributions of our work can be summarized as follows: (1) We are the first to propose an end-to-end argument-aware event representation-based event linking framework that unifies the modeling of both in-KB and out-of-KB scenarios; (2) To improve the performance, we propose a contrastive learning module, enhancing the learned embeddings of events and event mentions from both mention–mention and mention–candidate views; (3) Experimental results show that the proposed event linking framework improves accuracy by at least 3.59 and 21.5 in in-KB testing and out-of-KB testing, respectively, while its inference speed is more than 38 times faster than baselines, demonstrating its accuracy and efficiency.

Task Definition

Given a token sequence $\mathcal{T} = \{x_1, x_2, \ldots, x_i, \ldots, x_j, \ldots\}$, a contiguous token span $m = \{x_i, \ldots, x_j\} \subset \mathcal{T}$, and a KB Q, where x_i denotes a token, m denotes the target event mention, the objective of the event linking task is to predict the label $y \in \mathcal{E} \cup \{\text{NIL}\}$ for m, where $\mathcal{E}$ denotes the event set of Q, and NIL denotes the out-of-KB event. The decision is made based on whether m corresponds to an entry $e \in \mathcal{E}$ or refers to an unseen event instance not covered by Q like the previous study (Hsu et al. 2024).

The Framework SEFEL

As shown in Figure 2, SEFEL consists of four parts: (1) out-of-KB event sample generation; (2) event linking base module; (3) contrastive learning module; (4) optimization and inference. In this section, we will introduce them in detail.

Out-of-KB Event Sample Generation

To enhance the event linking model to make robust predictions not only for in-KB but also for out-of-KB events, we propose a method to generate samples for out-of-KB events to augment the training data. Inspired by (Hsu et al. 2024), we also explore to leverage the event arguments to construct samples for out-of-KB events. Unlike existing event argument extraction methods (Huang et al. 2022; Hsu et al. 2023a) that typically rely on heuristic rules or annotated data sets tailored to specific domains, limiting their abilities to handle events with complex semantics and diverse structures in open domain settings, we propose to use an LLM to perform event type identification and event argument extraction. Formally, given an event mention m in the input sentence $\mathcal{T}_m$, our goal is to identify its event type, then extract a set of event arguments $\mathcal{A}_m = \{a_1, a_2, \ldots, a_k\}$, where a_i denotes a salient semantic component such as time, location, and participants. The extraction process can be denoted as follows:

$$\mathcal{T}_m \xrightarrow{\text{LLM}} \mathcal{A}_m. \tag{1}$$

Leveraging the LLM's powerful contextual understanding ability, key semantic factors can be captured from $\mathcal{T}_m$. Note that this simple extraction method does not require annotated data, thereby it has strong domain generalization ability. Subsequently, we retain m and define a perturbation function on $\mathcal{A}_m$:

$$\mathcal{A}_m \xrightarrow{\text{LLM}} \tilde{\mathcal{A}}_m, \tag{2}$$

where $\tilde{\mathcal{A}}_m$ denotes the set of modified arguments produced by substituting one or more elements in $\mathcal{A}_m$ through LLM-guided editing. Next, we use the LLM to regenerate the full sentence $\tilde{\mathcal{T}}_m$ conditioned on m and $\tilde{\mathcal{A}}_m$:

$$(m, \tilde{\mathcal{A}}_m) \xrightarrow{\text{LLM}} \tilde{\mathcal{T}}_m. \tag{3}$$

The regenerated sentence $\tilde{\mathcal{T}}_m$ maintains grammatical correctness and structural consistency compared with $\mathcal{T}_m$, while introducing subtle semantic shifts that make it incompatible with any known events in the KB.

Event Linking Base Module

Inspired by the previous work (Ayoola et al. 2022), we propose an end-to-end event linking base module, which trains two transformer encoders jointly: one for the candidate event side and the other for the event mention side. For the candidate event side, we utilize the candidate event description as input. Next, we will construct an argument-aware event mention representation.

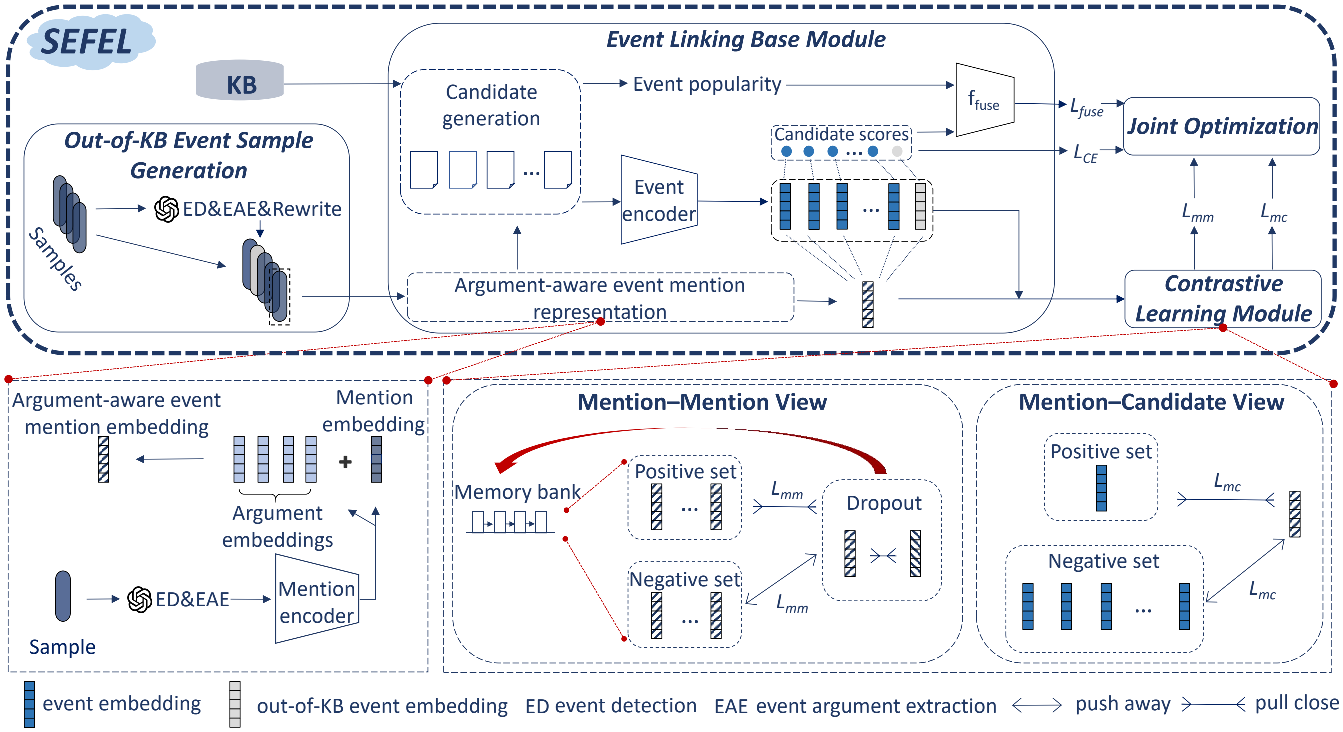

Figure 2: Overview of our framework SEFEL.

Argument-Aware Event Mention Representation. To obtain a more comprehensive event representation, building upon the event mention and its corresponding event arguments, we construct an argument-aware event mention representation. Specifically, for a given event mention m with its associated event arguments $\mathcal{A}_m$, we define the argument-aware event mention as $\mathcal{M} = m \cup \mathcal{A}_m$ to substitute the original event mention, better representing the entirety of an event. Let $\mathbf{h}_a$ denote the contextualized embedding of $a \in \mathcal{A}_m$. Then, we construct the embedding $\mathbf{h}_{\mathcal{M}}$ for $\mathcal{M}$ by combining the event mention representation with its event argument representations as follows:

$$\mathbf{h}_{\mathcal{M}} = \frac{1}{|\mathcal{A}_m| + 1}(\mathbf{h}_m + \sum_{a \in \mathcal{A}_m} \mathbf{h}_a). \tag{4}$$

This embedding encodes richer semantic information than the event mention alone, which may be ambiguous or incomplete, facilitating better alignment with the candidate event in the KB. Additionally, for m and $a \in \mathcal{A}_m$, we can obtain their corresponding candidate event sets through a dictionary-based retrieval strategy like previous studies (Fang et al. 2019; Li et al. 2022). Specially, for $\mathcal{M}$, we construct a candidate event set $\mathcal{E}_{\mathcal{M}}$ by combining the candidate event sets of m and all event arguments of $\mathcal{A}_m$.

Loss Function. To further align $\mathbf{h}_{\mathcal{M}}$ with the candidate event space, we map it into a shared semantic space through a linear transformation $f_1(\cdot)$ denoted by $\mathbf{z}_{\mathcal{M}} = f_1(\mathbf{h}_{\mathcal{M}})$. We map the candidate event embedding into the shared semantic space using a linear transformation $f_2(\cdot)$ denoted by $\mathbf{z}_{e_{\mathcal{M}}} = f_2(\mathbf{h}_{e_{\mathcal{M}}})$. To seamlessly incorporate out-of-KB event prediction into the unified scoring architecture, for each M, we add the out-of-KB event denoted by NIL as an additional candidate event to the corresponding candidate event set $\mathcal{E}_M$.

Event Popularity. Inspired by (Liu et al. 2021; Liu, Shen, and Yuan 2016), we leverage event popularity as a signal of the task of event linking. We define the prior probability $\hat{P}(e_{\mathcal{M}} \mid \mathcal{M})$ to evaluate the popularity of $e_{\mathcal{M}}$ given $\mathcal{M}$ based on the anchor link statistics from Wikipedia as follows:

$$\hat{P}(e_{\mathcal{M}} \mid \mathcal{M}) = \frac{count(\mathcal{M}, e_{\mathcal{M}})}{count(\mathcal{M})}, \tag{5}$$

where $count(\mathcal{M}, e_{\mathcal{M}})$ denotes the number of anchor links in Wikipedia in which either the event mention or any of its associated event argument link to $e_{\mathcal{M}}$, and $count(\mathcal{M})$ is the number of the event mention and it's associated event arguments occurring as the surface form of an anchor link. Note that $\hat{P}(\text{NIL} \mid \mathcal{M})$ is set to zero, which is consistent with the fact that out-of-KB event mentions do not appear as anchor links in the corpus.

Event Description. To effectively distinguish the gold-standard event from other retrieved candidates, we adopt a softmax-based scoring objective over candidate event descriptions. Given $\mathbf{z}_{\mathcal{M}}$ and $\mathcal{E}_{\mathcal{M}}$, we calculate the similarity between $\mathbf{z}_{\mathcal{M}}$ and each candidate event embedding via the dot product. A softmax is applied over these scores and the linking model can be trained by maximizing the likelihood of the correct candidate via the cross-entropy loss as follows:

$$\mathcal{L}_{ce}(\mathcal{M}, e_{\mathcal{M}}) = -y \log \frac{\exp(\mathbf{z}_{\mathcal{M}} \cdot \mathbf{z}_{\mathbf{e}_{\mathcal{M}}})}{\sum_{e'_{\mathcal{M}} \in \mathcal{E}_m} \exp(\mathbf{z}_{\mathcal{M}} \cdot \mathbf{z}'_{\mathbf{e}_{\mathcal{M}}})}, \quad (6)$$

where $y \in \{0, 1\}$ denotes the gold-standard label of the candidate event. If $e_{\mathcal{M}}$ is the correct corresponding event for $\mathcal{M}$, the value of y is 1; otherwise 0. For out-of-KB events, the candidate description score is given by the similarity between $\mathbf{z}_{\mathcal{M}}$ and $\mathbf{z}_{\text{NIL}}$.

To combine the strength of event popularity and event description, we define the fusion score as follows:

$$\omega(\mathcal{M}, e_{\mathcal{M}}) = f_{\text{fuse}}(\mathbf{z}_{\mathcal{M}} \cdot \mathbf{z}_{\mathbf{e}_{\mathcal{M}}},\ \hat{P}(e_{\mathcal{M}} \mid \mathcal{M})). \quad (7)$$

We adopt a linear layer $f_{\text{fuse}}(\cdot)$ to integrate the two components and produce the similarity score between $\mathbf{z}_{\mathcal{M}}$ and $\mathbf{z}_{e_{\mathcal{M}}}$. To optimize this event linking base module, we adopt a cross-entropy loss over $\omega(\mathcal{M}, e_{\mathcal{M}})$. The objective function is defined as follows:

$$\mathcal{L}_{\text{fuse}}(\mathcal{M}, e_{\mathcal{M}}) = -y \log \frac{\exp(\omega(\mathcal{M}, e_{\mathcal{M}}))}{\sum_{e'_{\mathcal{M}} \in \mathcal{E}_m} \exp(\omega(\mathcal{M}, e'_{\mathcal{M}}))}. \quad (8)$$

Consequently, the final fusion score for the out-of-KB event is computed using the same fusion mechanism. This design allows the end-to-end model to learn out-of-KB event behavior. Notably, it avoids the need for reliance on hand-crafted thresholds or external out-of-KB event classifiers and enables robust detection of out-of-KB mentions during inference. Thus, we define the base loss function as follows:

$$\mathcal{L}_{\text{base}}(\mathcal{M}, e_{\mathcal{M}}) = \alpha \cdot \mathcal{L}_{\text{ce}}(\mathcal{M}, e_{\mathcal{M}}) + \beta \cdot \mathcal{L}_{\text{fuse}}(\mathcal{M}, e_{\mathcal{M}}), \quad (9)$$

where α and β are weights of $\mathcal{L}_{\text{ce}}$ and $\mathcal{L}_{\text{fuse}}$, respectively.

Contrastive Learning Module

Contrastive learning is a very effective paradigm for representation learning, which utilizes both positive and negative pairs to model fine-grained semantic relationships. By minimizing the distance between semantically similar instances while maximizing the separation between dissimilar ones, it can enhance intra-class cohesion and inter-class discriminability. Motivated by these advantages, we propose the contrastive learning strategy from both mention-mention and mention-candidate views to enhance the event linking base module's ability to distinguish structurally similar but semantically different events. As the example shown in Figure 1, for documents $d3$ and $d5$, although they have the same event mention (i.e., *"evacuated"*) and similar event argument structure (i.e., location: *Novosibirsk*, time: *In 1942*), they refer to different events. $d3$ refers to the event *"World War II"* and $d5$ refers to the out-of-KB event.

Mention–Mention View. To facilitate efficient training with a large number of positive and negative pairs across batches and promote alignment of argument-aware event mention embeddings with identical event semantics while enhancing separability for distinct events, we employ a contrastive learning objective with dynamically constructed positive and negative pairs. Inspired by (He et al. 2020), we maintain a momentum-updated memory bank $\mathcal{B} = \{(\mathbf{z}_j, y_j)\}_{j=1}^{N}$, where $\mathbf{z}_j$ denotes an argument-aware event mention embedding, y_j denotes the corresponding event of $\mathbf{z}_j$, and N denotes the size of the memory bank. The memory bank serves as a first-in-first-out queue that stores some recent argument-aware event mention embeddings across training steps. For the current argument-aware mention embedding $\mathbf{z}_{\mathcal{M}}$, we first generate a correlated embedding $\mathbf{z}'_{\mathcal{M}}$ through independently sampled dropout masks, forming a positive pair representing the same event under different stochastic perturbations. Subsequently, both embeddings are pushed into $\mathcal{B}$ to serve as potential contrastive samples for future argument-aware event mentions. To perform contrastive learning, we define the following sets based on $\mathbf{z}_{\mathcal{M}}$ with its corresponding event $y_{\mathcal{M}}$:

- Positive set: $\mathcal{P}_{\mathbf{z}_{\mathcal{M}}} = \{\mathbf{z}_p \in \mathcal{B} \mid y_p = y_{\mathcal{M}}\}$.
- Negative set: $\mathcal{N}_{\mathbf{z}_{\mathcal{M}}} = \{\mathbf{z}_n \in \mathcal{B} \mid y_n \neq y_{\mathcal{M}}\}$.
- Full contrastive set : $\mathcal{Q}_{\mathbf{z}_{\mathcal{M}}} = \mathcal{P}_{\mathbf{z}_{\mathcal{M}}} \cup \mathcal{N}_{\mathbf{z}_{\mathcal{M}}}$.

The mention–mention contrastive loss is calculated as follows:

$$\mathcal{L}_{mm}(\mathcal{M}) = -\log \frac{\sum_{\mathbf{z}_p \in \mathcal{P}_{\mathbf{z}_{\mathcal{M}}}} \exp(\mathbf{z}_{\mathcal{M}} \cdot \mathbf{z}_p / \tau)}{\sum_{\mathbf{z}_n \in \mathcal{Q}_{\mathbf{z}_{\mathcal{M}}}} \exp(\mathbf{z}_{\mathcal{M}} \cdot \mathbf{z}_n / \tau)}, \quad (10)$$

where τ is a temperature hyperparameter. $\mathcal{Q}_{\mathbf{z}_{\mathcal{M}}}$ and $\mathcal{L}_{\text{mm}}(\mathcal{M})$ are initialized as $\varnothing$ and 0, respectively.

Mention–Candidate View. To improve the semantic alignment between argument-aware event mentions and their corresponding events while enhancing discrimination of similar candidates, we introduce a contrastive learning objective between argument-aware event mention embeddings and its candidate event embeddings. The contrastive loss encourages the argument-aware event mention embedding to be closer to its gold-standard event and further from the different candidate events. The mention–candidate contrastive loss is calculated as follows:

$$\mathcal{L}_{\text{mc}}(\mathcal{M}, e_{\mathcal{M}}) = -y \log \frac{\exp(\mathbf{z}_{\mathcal{M}} \cdot \mathbf{z}_{e_{\mathcal{M}}} / \tau)}{\sum_{e'_{\mathcal{M}} \in \mathcal{E}_m} \exp(\mathbf{z}_{\mathcal{M}} \cdot \mathbf{z}_{e'_{\mathcal{M}}} / \tau)}. \quad (11)$$

We define the final contrastive loss as the sum of the mention–mention contrastive loss and the mention–candidate contrastive loss as follows:

$$\mathcal{L}_{\text{ct}}(\mathcal{M}, e_{\mathcal{M}}) = \gamma \cdot \mathcal{L}_{\text{mm}}(\mathcal{M}, e_{\mathcal{M}}) + \delta \cdot \mathcal{L}_{\text{mc}}(\mathcal{M}, e_{\mathcal{M}}), \quad (12)$$

where γ and δ are weights of $\mathcal{L}_{\text{mm}}$ and $\mathcal{L}_{\text{mc}}$, respectively.

Optimization and Inference

To improve performance, we combine all the loss components introduced above and jointly optimize for training:

$$\mathcal{L} = \sum_{\mathcal{M} \in \mathbb{M}} \sum_{e_{\mathcal{M}} \in \mathcal{E}_{\mathcal{M}}} \zeta \cdot \mathcal{L}_{\text{base}}(\mathcal{M}, e_{\mathcal{M}}) + \eta \cdot \mathcal{L}_{\text{ct}}(\mathcal{M}, e_{\mathcal{M}}), \quad (13)$$

where $\mathbb{M}$ denotes the argument-aware event mention set of the training set. ζ and η are weights of the base loss and contrastive loss, respectively. These parameters are selected based on validation data to balance semantic discrimination and generalization. During training, the model learns to cluster argument-aware event mentions with their corresponding events while separating different events, ultimately improving the linking performance.

Data set	Train	Valid	Test	
			In-KB	Out-of-KB
Wikipedia	66425	16692	19267	-
NYT	-	-	769	993

Table 1: Statistics of the two data sets.

Experiments

Experimental Setting

Data Sets. We conduct experiments on two publicly available data sets: (1) Wikipedia data set; (2) New York Times (NYT) data set, used in previous event linking studies (Yu et al. 2023; Hsu et al. 2024). These two data sets are aligned with a structured event KB constructed from English Wikipedia pages with titles. The Wikipedia data set is automatically constructed using internal hyperlinks in Wikipedia, serving as the in-KB evaluation set. Table 1 shows the detailed statistics of the data sets. Specifically, if the anchor text of a link points to a Wikipedia page that is categorized under the "Event" type according to the FIGER schema (Ling and Weld 2012), it is considered a valid event mention. As a result, all instances in this data set are guaranteed to link to entries that exist within the KB. The NYT data set serves as an out-of-KB evaluation set, which consists of event mentions manually annotated from $2,500$ lead paragraphs of the New York Times Annotated Corpus. A substantial portion of these event mentions refer to real-world events absent from the KB, and NIL is their gold-standard label. We make the data sets and source code used in this paper publicly available for future research.

Evaluation Measures. Following previous event linking studies (Yu et al. 2023; Hsu et al. 2024), we adopt the same ranking metric accuracy to evaluate all methods.

Setting Details. Mention and candidate encoders are both based on RoBERTa-base. The weights of the cross-entropy loss for description similarity α, fusion scoring loss β, mention–mention contrastive loss γ, mention–candidate contrastive loss δ, base module loss function ζ, contrastive loss η, memory bank size $\mathcal{N}$, temperature parameter τ are set to 0.01, 1, 0.001, 0.01, 512, 0.07, respectively. For event argument extraction and out-of-KB sample generation, we use LLaMA-3.1-70B-Instruct as the LLM. The generated out-of-KB event training instances constitute approximately 1% of the total training data.

Effectiveness Study

We compare SEFEL with the following six baselines. BM25 retrieves relevant candidate events for the given event mention in the KB by using term-based matching and TF–IDF-based ranking. GENRE (Cao et al. 2021) formulates the event linking task as a conditional generation problem. It generates the target event directly by using a sequence-to-sequence Transformer trained on mention–title pairs, when provided with sentences containing event mentions. BLINK

Method	*Wikipedia Test*			*NYT Test*		
	All	*Verb*	*Noun*	*All*	*Verb*	*Noun*
BM25	9.72	13.08	6.36	3.69	3.18	5.19
GENRE	76.04	71.76	80.32	–	–	–
BLINK	78.74	78.12	79.36	27.13	29.24	20.74
EveLink	79.00	78.07	79.93	32.03	34.34	25.13
ArgEveLink	80.05	79.47	80.62	55.40	59.90	41.99
GPT-4o-mini	79.30	81.94	76.66	30.55	33.90	20.37
SEFEL	**83.64**	**85.99**	**81.29**	**76.90**	**79.61**	**68.85**

Table 2: Performance on the task of event linking. GPT-4o-mini is run via its official API with our constructed prompts. Other results of baselines are taken from (Hsu et al. 2024).

(Wu et al. 2020) first adopts a bi-encoder to compute dense embeddings of the event mention context and all candidate events, retrieving top candidate events via embedding similarity. Next, it uses a cross-encoder to jointly encode the input sentence and the candidate event description to rerank the retrieved candidate events. EveLINK (Yu et al. 2023) extends BLINK by enriching the input with named entities as supplementary tokens. ArgEveLINK (Hsu et al. 2024) builds upon EveLINK by incorporating additional contextual information derived from extracted event structures. Notably, it relies on traditional extraction pipelines—such as syntactic parsing and rule-based heuristics—to identify the event type and associated arguments from the raw text. These extracted elements are then encoded using predefined role-specific tokens and integrated into the model input. GPT-4o-mini performs event linking by leveraging an instruction-based reranking mechanism. It takes an event mention, its contextual sentence, and a list of candidate Wikipedia event titles as input, and is prompted to identify the most semantically compatible candidate. We evaluate above methods on the Wikipedia and NYT test set. It should be noted that following previous works (Yu et al. 2023; Hsu et al. 2024), we do not evaluate GENRE on the NYT test set, as its prediction probability is unclear. As shown in Table 2, the experimental results show that SEFEL consistently outperforms all baseline methods on both data sets across different event mention types (i.e., verbs and nouns). Specifically, for the in-KB setting, i.e., Wikipedia test data, SEFEL achieves exceeds the best basline method (i.e., ArgEveLink) by 3.59 percentages. For the out-of-KB setting, i.e., NYT test set, SEFEL exceeds the best baseline method (i.e., ArgEveLink) by 21.5 percentages, despite NYT contains more challenging event mentions, which have no corresponding events in the KB.

Efficiency Study

In addition to accuracy, the linking efficiency is also a critical consideration for deploying event linking systems in real-world scenarios, particularly when operating at scale or under time constraints. To evaluate the execution time of the linking process, we compare SEFEL against representative

Data set	*Method*	*Time (s)*
Wikipedia Test	ArgEveLINK	24612
	GPT-4o-mini	50293
	SEFEL	**446**
NYT Test	ArgEveLINK	2250
	GPT-4o-mini	4596
	SEFEL	**59**

Table 3: Efficiency study over Wikipedia and NYT test sets.

baselines (i.e., ArgEveLINK and GPT-4o-mini) using wall-clock time on both the Wikipedia and NYT test sets. Existing baselines BLINK, EveLINK, and ArgEveLINK build upon the BLINK retrieve-and-rerank framework. Since these methods share the same underlying architecture, we use ArgEveLINK with the best accuracy score among them as the representative baseline for execution time comparisons. All efficiency experiments were conducted in a unified environment with 4×RTX A6000 GPUs. As shown in Table 3, SEFEL requires significantly less linking time while achieving higher linking accuracy. Empirical evaluations in a shared environment demonstrate that SEFEL is nearly 40 times faster than ArgEveLINK and nearly 80 times faster than GPT-4o-mini. This efficiency gain arises from our lightweight encoding scheme and contrastive training strategy, which enable faster similarity calculation and candidate event ranking without sacrificing performance. Additionally, we implement a parallelized version of ArgEveLINK. However, even with parallelization, its efficiency remains substantially lower than SEFEL, with 72.5 accuracy and 6492 seconds on Wikipedia Test under the same environment.

Experimental Analysis of Argument-aware Event Mention Representation

To verify the effectiveness of argument-aware event mention representation, we present two variants of SEFEL in Table 4: (1) $\text{SEFEL}_{\text{Uni+Tag}}$. It utilizes the traditional event argument extraction method used by (Hsu et al. 2024) including Unist (Huang et al. 2022) and TagPrime (Hsu et al. 2023a) to replace the LLM ; and (2) SEFEL-EAE. It removes extracted event arguments entirely and relies solely on the event mention and its surrounding context. From the results shown in Table 4, it can be seen that SEFEL / $\text{SEFEL}_{\text{Uni+Tag}}$ outperforms SEFEL-EAE on both Wikipedia and NYT test sets, which demonstrates that event arguments can indeed improve the linking performance. Furthermore, SEFEL outperforms $\text{SEFEL}_{\text{Uni+Tag}}$, which demonstrates the effectiveness of our LLM-based event extraction method compared with traditional event argument extraction methods on the task of event linking. This may be attributed to the fact that the LLM extractor better captures semantically rich and context-sensitive arguments, which are crucial for constructing high-quality argument-aware mention embedding.

Ablation Study

To verify the contribution of different components in our proposed framework, we conduct ablation studies by removing each part separately and testing the performance of remaining parts on the Wikipedia and NYT test sets. From the results shown in Table 5, we can obtain some observations: (1) without the event description information, SEFEL's performance exhibits a significant decline on Wikipedia test set, which indicates the event description information is more applicable to in-KB testing; (2) without the event popularity information, SEFEL declines more significantly on the NYT test set, which indicates the event popularity information is more applicable to out-of-KB testing; (3) without the contrastive learning module's mention-mention view or mention-candidate view, SEFEL's performance declines on both data sets, which demonstrates the importance of contrastive supervision in improving the embeddings of events and event mentions to enhance the linking performance.

Method	*Wikipedia Test*			*NYT Test*		
	All	*Verb*	*Noun*	*All*	*Verb*	*Noun*
SEFEL	**83.64**	**85.99**	**81.29**	**76.90**	**79.61**	**68.85**
$\text{SEFEL}_{\text{Uni+Tag}}$	83.28	85.27	80.85	62.43	66.72	49.66
SEFEL-EAE	82.22	84.62	79.83	56.53	61.49	41.76

Table 4: Performance of different variants of SEFEL for analysis of argument-aware event mention representation.

Ablations	*Wikipedia Test*			*NYT Test*		
	All	*Verb*	*Noun*	*All*	*Verb*	*Noun*
SEFEL	**83.64**	**85.99**	**81.29**	**76.90**	**79.61**	**68.85**
w/o event description	58.64	59.37	57.92	76.84	79.3	68.62
w/o event popularity	82.28	84.91	79.65	56.92	61.87	42.21
w/o mention-mention view	82.47	85.09	79.84	74.63	76.95	67.72
w/o mention-candidate view	82.9	84.86	80.94	73.61	75.82	67.04

Table 5: Ablation study on the event linking task.

Parameter Study

To investigate the robustness of SEFEL, we conduct sensitivity analysis to better understand the impact of parameter τ (i.e., the temperature coefficient used across all contrastive losses) on SEFEL's performance on the Wikipedia and NYT test sets. The default value used in reported experiments is $\tau = 0.07$. From the trend plotted in Figure 3, the accuracy on the Wikipedia test set fluctuates by less than 0.62%, while the accuracy on the NYT test set fluctuates by less than 0.06%, it can be seen that the performance of SEFEL is very stable on both data sets, and is insensitive to τ.

Related Work

Event Linking

In recent years, Yu et al. (2023) formally defined event linking as aligning textual event mentions with corresponding

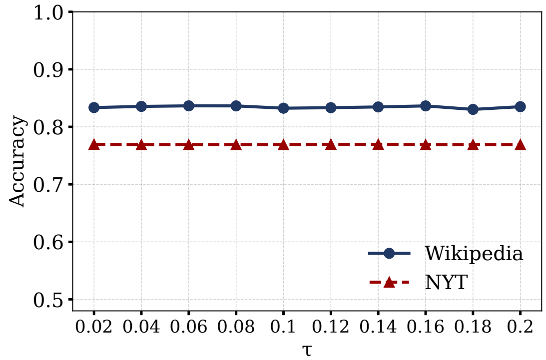

Figure 3: Parameter sensitivity analysis with varied temperature τ.

event nodes in a structured KB (e.g., Wikipedia). Subsequently, Pratapa, Gupta, and Mitamura (2022) introduced multilingual data sets for this task, (Hsu et al. 2024) incorporated event arguments to enhance the performance of the event linking task. All of these methods adopt a two-stage retrieve-and-rank framework: a bi-encoder retrieved candidate events and then a computationally intensive cross-encoder for fine-grained ranking (Wu et al. 2020). This dependence on cross-encoders creates a critical bottleneck, since their pairwise computation requirements lead to inefficiencies that hinder large-scale open-domain deployment for the event linking task. In contrast, SEFEL eliminates the reranking process entirely. By using a lightweight bi-encoder architecture enhanced with event argument fusion and contrastive supervision, we perform all candidate event scoring in a single forward pass. This end-to-end method significantly improves linking efficiency while maintaining a high linking accuracy.

Event Argument Extraction

Event argument extraction (EAE) is a fundamental subtask of event extraction that aims to identify and classify argument roles associated with event triggers. Traditional approaches typically formulated EAE as a sequence labeling or span classification task, using neural models to predict argument boundaries and roles based on sentence-level context (Li, Ji, and Han 2021; Zeng, Zhan, and Ji 2022; Hsu et al. 2023b; He, Hu, and Tang 2023). Recent studies explored joint modeling of event triggers and arguments, hierarchical representations, and sentence-level reasoning to improve accuracy. With the rise of pre-trained LLMs, several works demonstrated that LLMs can serve as effective zero-shot or few-shot annotators through innovative prompting techniques. Notably, (Chen et al. 2024a) proposed a prompting-based method to guide the LLM in extracting structured event argument tuples directly from text, achieving competitive performance without extensive labeled data. Similarly, our framework also utilizes an LLM to extract event arguments automatically. Based on these arguments, we integrate them with the event mention to generate an argument-aware event representation, enhancing the performance of the event linking task.

Contrastive Learning

Contrastive learning was successfully applied in many applications, such as event extraction (Wang et al. 2021), event detection (Ren et al. 2023), event coreference resolution (Hsu and Horwood 2022), and other KB-ralated works (e.g., multimodal entity linking (Hu et al. 2025), KB completion (Ko et al. 2025), KB-enhanced recommendation systems (Yang et al. 2022), OKB canonicalization (Yang et al. 2025), and temporal KB reasoning (Chen et al. 2024b). In this paper, we apply the contrastive learning to solve the event linking task successfully.

Entity Linking

Entity linking, the task of aligning entity mentions with their corresponding entities in the KB, was extensively studied (Shen et al. 2021). Recent advances was shifted toward neural architectures, particularly the retrieve-and-rerank framework exemplified by BLINK (Wu et al. 2020). In BLINK, a bi-encoder retrieved top-k candidates by encoding mentions and entities separately, followed by a cross-encoder that jointly processes mention–candidate pairs for fine-grained scoring. The generation-based GENRE model (Cao et al. 2021) proposed an alternative formulation by framing entity linking as a sequence-to-sequence generation task, where the entity title is generated autoregressively given the mention context. Recent works GSM (Cheng et al. 2024) and XMRED (Jiang et al. 2024) further explored joint learning of linking and grounding through type constraints or global coherence objectives. In addition to linking mentions to in-KB entities, a critical challenge in entity linking lies in detecting out-of-KB cases, which do not have corresponding entities in the KB. Accurate out-of-KB mention detection is essential for maintaining the entity linking system's robustness in open-domain applications (Dong et al. 2023).

Conclusion

In this paper, we propose an end-to-end argument-aware event representation-based event linking framework via contrastive learning, unifying the modeling of both in-KB and out-of-KB scenarios. Extensive experiments have demonstrated the effectiveness and efficiency of our proposed framework against many baseline methods.

Acknowledgments

The work is partially supported by the National Key Research and Development Program of China (2024YFF0617702), the National Natural Science Foundation of China (Nos. U22A2025, 62402097, 62232007, U23A20309), the Joint Funds of Natural Science Foundation of Liaoning Province (2023-BSBA-132), and 111 Project (No. B16009), the Ant Group Research Program (2025021900003).

References

Ayoola, T.; Tyagi, S.; Fisher, J.; Christodoulopoulos, C.; and Pierleoni, A. 2022. ReFinED: An Efficient Zero-shotcapable Approach to End-to-End Entity Linking. In *NAACL*, 209–220.

Cao, N. D.; Izacard, G.; Riedel, S.; and Petroni, F. 2021. Autoregressive Entity Retrieval. In *ICLR*.

Chen, R.; Qin, C.; Jiang, W.; and Choi, D. 2024a. Is a large language model a good annotator for event extraction? In *AAAI*, 16, 17772–17780.

Chen, W.; Wan, H.; Wu, Y.; Zhao, S.; Cheng, J.; Li, Y.; and Lin, Y. 2024b. Local-Global History-Aware Contrastive Learning for Temporal Knowledge Graph Reasoning. In *ICDE*, 733–746.

Cheng, X.; Wang, Y.; Fan, Y.; Guo, J.; Zhang, R.; and Bi, K. 2024. GSM-EL: A Generalizable Symbol-Manipulation Approach for Entity Linking. *IEEE Transactions on Knowledge & Data Engineering*, (01): 1–14.

Dong, H.; Chen, J.; He, Y.; Liu, Y.; and Horrocks, I. 2023. Reveal the unknown: out-of-knowledge-base mention discovery with entity linking. In *CIKM*, 452–462.

Fang, Z.; Cao, Y.; Li, Q.; Zhang, D.; Zhang, Z.; and Liu, Y. 2019. Joint entity linking with deep reinforcement learning. In *WWW*, 438–447.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 9729–9738.

He, Y.; Hu, J.; and Tang, B. 2023. Revisiting Event Argument Extraction: Can EAE Models Learn Better When Being Aware of Event Co-occurrences? In *ACL*, 12542–12556.

Hoffart, J.; Altun, Y.; and Weikum, G. 2014. Discovering emerging entities with ambiguous names. In *WWW*, 385–396.

Hoffart, J.; Milchevski, D.; Weikum, G.; Anand, A.; and Singh, J. 2016. The knowledge awakens: Keeping knowledge bases fresh with emerging entities. In *WWW*, 203–206.

Hsu, B.; and Horwood, G. 2022. Contrastive Representation Learning for Cross-Document Coreference Resolution of Events and Entities. In *NAACL*, 3644–3655.

Hsu, I.-H.; Huang, K.-H.; Zhang, S.; Cheng, W.; Natarajan, P.; Chang, K.-W.; and Peng, N. 2023a. TAGPRIME: A Unified Framework for Relational Structure Extraction. In *ACL*, 12917–12932.

Hsu, I.-H.; Xie, Z.; Huang, K.-H.; Natarajan, P.; and Peng, N. 2023b. AMPERE: AMR-Aware Prefix for Generation-Based Event Argument Extraction Model. In *ACL*, 10976–10993.

Hsu, I.-H.; Xue, Z.; Pochhi, N.; Bansal, S.; Natarajan, P.; Srinivasa, J.; and Peng, N. 2024. Argument-Aware Approach To Event Linking. In *Findings of ACL*, 12769–12781.

Hu, Z.; Gutiérrez-Basulto, V.; Li, R.; and Pan, J. Z. 2025. Multi-level Matching Network for Multimodal Entity Linking. In *KDD*, 508–519.

Huang, J. Y.; Li, B.; Xu, J.; and Chen, M. 2022. Unified Semantic Typing with Meaningful Label Inference. In *NAACL*, 2642–2654.

Jiang, J.-Y.; Chang, W.-C.; Zhang, J.; Hsieh, C.-J.; and Yu, H.-F. 2024. Entity disambiguation with extreme multi-label ranking. In *WWW*, 4172–4180.

Ko, Y.; Yang, H.; Kim, T.; and Kim, H. 2025. Subgraph-Aware Training of Language Models for Knowledge Graph Completion Using Structure-Aware Contrastive Learning. In *WWW*, 72–85.

Li, R.; Wang, Z.; Tran, S.; Xia, L.; and Du, X. 2024. Meqa: A benchmark for multi-hop event-centric question answering with explanations. *NeurIPS*, 126835–126862.

Li, S.; Ji, H.; and Han, J. 2021. Document-Level Event Argument Extraction by Conditional Generation. In *NAACL*, 894–908.

Li, Y.; Shen, W.; Gao, J.; and Wang, Y. 2022. Community question answering entity linking via leveraging auxiliary data. *IJCAI*, 2145–2151.

Ling, X.; and Weld, D. 2012. Fine-grained entity recognition. In *AAAI*, 94–100.

Liu, Y.; Chen, H.; Shen, W.; and Chen, J. 2023. Low-resource personal attribute prediction from conversations. In *AAAI*, 4507–4515.

Liu, Y.; Shen, W.; Wang, Y.; Wang, J.; Yang, Z.; and Yuan, X. 2021. Joint open knowledge base canonicalization and linking. In *SIGMOD*, 2253–2261.

Liu, Y.; Shen, W.; Yao, Z.; Wang, J.; Yang, Z.; and Yuan, X. 2020. Named entity location prediction combining twitter and web. *IEEE Transactions on Knowledge and Data Engineering*, 33(11): 3618–3633.

Liu, Y.; Shen, W.; and Yuan, X. 2016. Deola: A System for Linking Author Entities in Web Document with DBLP. In *CIKM*, 2449–2452.

Pratapa, A.; Gupta, R.; and Mitamura, T. 2022. Multilingual Event Linking to Wikidata. In *Proceedings of the Workshop on Multilingual Information Access (MIA)*, 37–58.

Ren, J.; Peng, H.; Jiang, L.; Liu, Z.; Wu, J.; Yu, Z.; and Yu, P. S. 2023. Uncertainty-guided boundary learning for imbalanced social event detection. *IEEE Transactions on Knowledge and Data Engineering*, 36(6): 2701–2715.

Shen, W.; Li, Y.; Liu, Y.; Han, J.; Wang, J.; and Yuan, X. 2021. Entity linking meets deep learning: Techniques and solutions. *IEEE Transactions on Knowledge and Data Engineering*, 35(3): 2556–2578.

Shen, W.; Liu, Y.; and Wang, J. 2018. Predicting Named Entity Location Using Twitter . In *ICDE*, 161–172.

Shen, W.; Wang, J.; and Han, J. 2014. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, 27(2): 443–460.

Wang, S.; Xu, X.; Zhang, X.; Wang, Y.; and Song, W. 2022. Veracity-aware and event-driven personalized news recommendation for fake news mitigation. In *WWW*, 3673–3684.

Wang, Z.; Wang, X.; Han, X.; Lin, Y.; Hou, L.; Liu, Z.; Li, P.; Li, J.; and Zhou, J. 2021. CLEVE: Contrastive Pre-training for Event Extraction. In *ACL*, 6283–6297.

Wu, L.; Petroni, F.; Josifoski, M.; Riedel, S.; and Zettlemoyer, L. 2020. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. In *EMNLP*, 6397–6407.

Yang, Y.; Huang, C.; Xia, L.; and Li, C. 2022. Knowledge Graph Contrastive Learning for Recommendation. In *SIGIR*, 1434–1443.

Yang, Y.; Shen, W.; Shu, J.; Liu, Y.; Curry, E.; and Li, G. 2025. CMVC+: a Multi-View Clustering Framework for Open Knowledge Base Canonicalization via Contrastive Learning. *IEEE Transactions on Knowledge and Data Engineering*.

Yu, X.; Yin, W.; Gupta, N.; and Roth, D. 2023. Event Linking: Grounding Event Mentions to Wikipedia. In *EACL*, 2671–2680.

Zeng, Q.; Zhan, Q.; and Ji, H. 2022. EA2E: Improving Consistency with Event Awareness for Document-Level Argument Extraction. In *Findings of NAACL*, 2649–2655.